CreateProcess-02

Parent process has explicit trust from child

Sean Barnum, Cigital, Inc. [vita[1]]

2007-03-20

Part "Original Cigital Coding Rule in XML"

Mime-type: text/xml, size: 4800 bytes

<table>
<tr><td><b>Attack Category</b></td><td><ul><li>Privilege Exploitation</li></ul></td></tr>
<tr><td><b>Vulnerability Category</b></td><td><ul><li>Process management</li><li>Unconditional</li></ul></td></tr>
<tr><td><b>Software Context</b></td><td><ul><li>Threads and Processes</li></ul></td></tr>
<tr><td><b>Location</b></td><td></td></tr>
<tr><td><b>Description</b></td><td>Parent process has explicit trust from child.
In calling CreateProcess(), you receive a thread handle and process handle to the child process. These handles are sufficient to allow the parent to completely rewrite the child using functions such as WriteProcessMemory. An interesting result of this is that the child process must trust your parent process. A parent always has the ability to rewrite the child.</td></tr>
<tr><td><b>APIs</b></td><td><table>
<tr><th>FunctionName</th><th>Comments</th></tr>
<tr><td>CreateProcess</td><td></td></tr>
<tr><td>CreateProcessA</td><td>ANSII implementation</td></tr>
<tr><td>CreateProcessW</td><td>Unicode implementation</td></tr>
<tr><td>CreateProcessAsUser</td><td></td></tr>
<tr><td>CreateProcessWithLogonW</td><td></td></tr>
</table></td></tr>
<tr><td><b>Method of Attack</b></td><td>Just a warning: you must trust your parent. Parent can overwrite process image.</td></tr>
<tr><td><b>Exception Criteria</b></td><td></td></tr>
<tr><td><b>Solutions</b></td><td><table>
<tr><th>Solution Applicability</th><th>Solution Description</th><th>Solution Efficacy</th></tr>
<tr><td>Child process handles sensitive data or operations that parent should not have access to.</td><td>Child process should not do anything that it would be catastrophic for the parent to have access to.</td><td>Effective, but may conceivably mean some things can't be done.</td></tr>
</table></td></tr>
</table>

[1] http://buildsecurityin.us-cert.gov/bsi/about_us/authors/35-BSI.html (Barnum, Sean)

Signature Details	BOOL CreateProcess(LPCTSTR lpApplicationName, LPTSTR lpCommandLine, LPSECURITY_ATTRIBUTES lpProcessAttributes, LPSECURITY_ATTRIBUTES lpThreadAttributes, BOOL bInheritHandles, DWORD dwCreationFlags, LPVOID lpEnvironment, LPCTSTR lpCurrentDirectory, LPSTARTUPINFO lpStartupInfo, LPPROCESS_INFORMATION lpProcessInformation); BOOL CreateProcessAsUser(HANDLE hToken, LPCTSTR lpApplicationName, LPTSTR lpCommandLine, LPSECURITY_ATTRIBUTES lpProcessAttributes, LPSECURITY_ATTRIBUTES lpThreadAttributes, BOOL bInheritHandles, DWORD dwCreationFlags, LPVOID lpEnvironment, LPCTSTR lpCurrentDirectory, LPSTARTUPINFO lpStartupInfo, LPPROCESS_INFORMATION lpProcessInformation); BOOL CreateProcessWithLogonW(LPCWSTR lpUsername, LPCWSTR lpDomain, LPCWSTR lpPassword, DWORD dwLogonFlags, LPCWSTR lpApplicationName, LPWSTR lpCommandLine, DWORD dwCreationFlags, LPVOID lpEnvironment, LPCWSTR lpCurrentDirectory, LPSTARTUPINFOW lpStartupInfo, LPPROCESS_INFORMATION lpProcessInfo);

Examples of Incorrect Code	`/* In child process... */`
	`doSomethingParentShouldNotHaveAccessTo();`

Examples of Corrected Code	`/* In child process... */`
	`/* Should not include functionality if parent having access would be a serious problem. */`

Source Reference	• http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dllproc/base/processes_and_threads.asp[2]

Recommended Resource		
Discriminant Set	**Operating System**	• Windows
	Languages	• C • C++

Cigital, Inc. Copyright

The Build Security In (BSI) portal is sponsored by the U.S. Department of Homeland Security (DHS), National Cyber Security Division. The Software Engineering Institute (SEI) develops and operates BSI. DHS funding supports the publishing of all site content.

[1] mailto:copyright@cigital.com